CHAPTER

4

MANAGING WINDOWS 2000 FILE SYSTEMS AND STORAGE

After reading this chapter and completing the exercises, you will be able to:

- Understand basic and dynamic storage
- List the drive configurations supported by Windows 2000
- Distinguish the FAT, FAT32, and NTFS file systems
- Describe permissions, sharing, and other security issues related to file systems
- Understand drive, volume, and partition maintenance and administration under Windows 2000

The Windows 2000 file storage system offers versatile disk management. With the addition of dynamic storage, Windows 2000 is able to support large disk volumes, provide fault tolerance, control access, and offer high performance. By retaining support for previous disk configurations and adding support for FAT32, Windows 2000 is fully capable of operating within a multiboot system. This chapter discusses basic and dynamic storage methods, file systems and drive configurations supported by Windows 2000, and all of the built-in tools used for disk maintenance.

FILE STORAGE BASICS

Windows 2000 supports two types of storage: basic and dynamic. Basic storage is the storage method with which most DOS and Windows NT users are familiar, and centers around partitioning a physical disk. Dynamic storage is a new method supported only by Windows 2000, and is not based around partitions, but is centered on volumes. A **volume** is a portion of one or more hard disks that is combined into a single logical structure, formatted with a single file system, and accessed via a single drive letter or mount point.

Basic Storage

Basic storage is the industry standard or traditional method of dividing a hard drive into partitions. A **partition** is a logical division of the physical space on a hard drive. Each partition can be formatted with a different file system. Partitions must be formatted before they can be used by an operating system.

There are two types of partitions: primary and extended. A single hard drive can host up to four primary partitions or it can host up to three primary partitions and a single extended partition. A **primary partition** is a type of partition on a basic disk that can be marked active, whereas an **extended partition** can be divided into logical drives. Only primary partitions and logical drives can be formatted with a file system. Under Windows 2000 Professional, the total number of formatted partitions cannot exceed 32 on a single physical drive. Thus, a single hard drive can appear as one or more accessible or usable drives (that is, after the partition is properly formatted).

A primary partition can be marked as the **active partition**. This informs the computer's BIOS to look for operating system booting information on that partition. Only primary partitions can be active and only a single partition can be active at a time. The active partition does not have to be the first partition on the drive.

Volumes, in the basic storage type, are 2 to 32 partitions combined into a single logical structure formatted with a single file system. **Volume sets** can be extended simply by adding another partition. However, volume sets can be reduced in size only by breaking the set and creating a new set. The act of breaking the set destroys (or at least makes inaccessible) all data stored on the volume. A volume set can span multiple partitions on one or more physical drives. A volume set is interacted with via the operating system through a single drive letter and provides no fault tolerance. If a single drive or partition in a volume set fails, all data in the set is destroyed.

Typically, you'll want to create partitions or volumes as large as the operating system, hardware, and file system will allow. Under Windows 2000, those file systems and sizes are:

- FAT: 4 GB
- FAT32: 32 GB
- NTFS: 2 TB

Each formatted partition or volume set is assigned a drive letter. The letters A and B are typically reserved for floppy drives, but the letters C through Z can be used for hard-drive-hosted formatted partitions/volumes. Thus, only 24 formatted partitions can be accessed from Windows 2000. In most situations, this limitation does not impose a system restriction.

Basic storage supports a wide range of disk configurations, from single formatted partitions (often called drives or logical drives) to RAID 5 volumes. The main difference between basic storage and dynamic storage is that basic storage disk structures require a system reboot when changed.

Windows 2000 supports this traditional method of storage for backward compatibility with older operating systems. In other words, Windows 2000 can take control of drive configurations (discussed later in this chapter) from previous operating systems (Windows NT, 95, 98, and DOS), if the structure conforms to the current restrictions of the file systems they host and the hosted file system is supported by Windows 2000. Supported file systems include **FAT** (file allocation table, the 16-bit file system originally introduced with DOS), **FAT32** (the 32-bit FAT file system), and **NTFS** (**New Technology File System**, the preferred native file system of Windows 2000). However, Windows 2000 no longer supports the creation of basic storage type drive structures beyond single formatted partitions; it can manage only existing structures.

Windows 2000 can be installed only onto basic storage type partitions. There are two partitions associated with Windows 2000: the system partition and the boot partition. Please take careful note of their descriptions, because, in our opinion, they are counterintuitive. The **system partition** is the active partition where the boot files required to display the boot menu and initiate the booting of Windows 2000 are stored. The **boot partition** hosts the main Windows 2000 system files and is the initial default location for the paging file. The boot partition can be the same partition as the system partition, or it can be any other partition (or logical drive in an extended partition) on any drive hosted by the computer. Neither the system partition nor the boot partition can be a member of a volume set or stripe set. They both can be the source or original partition/drive in a disk mirror or disk duplexing configuration. The drive letters of the system partition and boot partition cannot be changed.

TIP

Once Windows 2000 is installed, the boot partition drive can be transformed into a dynamic storage device, but the system partition host must remain a basic storage device.

Dynamic Storage

Dynamic storage is a new type of storage technique (Microsoft documentation labels it as a new standard) that does not use partitions. Instead, this method views an entire physical hard drive as a single entity, labeled as a volume. This storage method offers drive structures from simple volumes (entire hard drives as a single formatted entity) to fully fault tolerant RAID 5 configurations. The main difference between dynamic storage and basic storage is that dynamic storage structures can be expanded on the fly without rebooting Windows 2000. Furthermore, only Windows 2000 can access data on dynamic storage volumes. No other operating system, including Windows 95, 98, or NT on a multiboot system, can access dynamic volumes.

New drives (including existing drives with all partitions deleted) can be transformed into dynamic storage hosts via a selection wizard. This wizard is launched when you access the **Disk Management** tool (Start, Programs, Administrative Tools, Computer Management, Storage, Disk Management) and a physical hard drive is present with no predefined partitions. This wizard only appears the first time Disk Management is accessed after booting that follows the addition of a new drive or the deletion of all partitions on a drive. You are prompted whether to enable dynamic storage.

Existing drives with partitions can be upgraded to dynamic storage by using the Upgrade to Dynamic Disk command. Windows 2000 Professional supports the following drive configurations:

- ***Simple volume***: All or part of a single drive. Does not provide any fault tolerance. NTFS volumes can be extended; FAT and FAT32 volumes cannot be extended.
- ***Spanned volume***: Two or more (up to 32) parts of one or more drives or two or more entire drives; the elements of the spanned volume do not have to be equal in size. Data is written to the first drive in the volume until it is full, then data is written to the next drive. This is also called an extended volume and does not provide any fault tolerance. If one partition or drive in the set fails, all data is lost. Spanned volumes cannot be part of a striped volume or a mirrored volume. NTFS spanned volumes can be extended; FAT and FAT32 spanned volumes cannot be extended. The system partition/volume and boot partition/volume cannot be extended. Volume sets can be reduced in size only by breaking the set and creating a new set. The act of breaking the set destroys all data stored on the volume.
- ***Striped volume**:* Two or more (up to 32) parts of one or more drives or two or more (up to 32) entire drives. Data is written to all drives in equal amounts (in 64 KB units) to spread the workload and improve performance. Each part or drive must be roughly equal in size. This storage scheme does not provide any fault tolerance—if one partition or drive in the set fails, all data is lost. Striped volumes cannot be mirrored or extended.

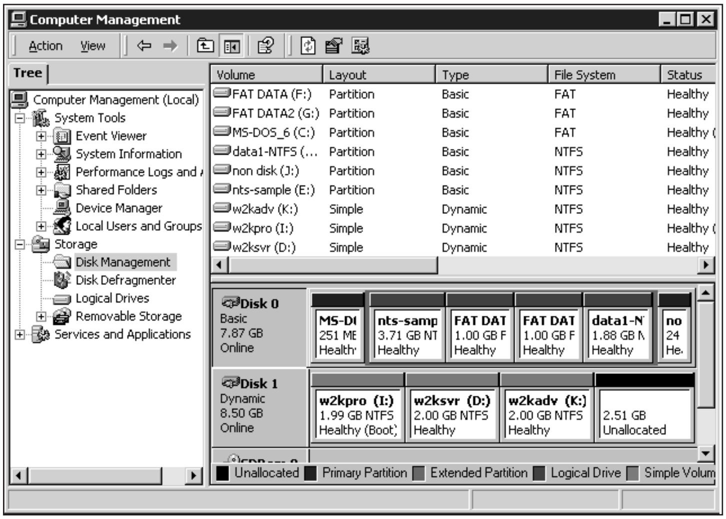
Windows 2000 Server also supports the following fault tolerant drive configurations:

- ***Mirrored volume**:* A single volume is duplicated onto another volume on a different hard drive. This storage scheme provides fault tolerance. In Windows NT, a mirror on a drive hosted by a different drive controller was called duplexing, but this distinction no longer is used in Windows 2000.
- ***RAID 5 volume**:* Three or more (up to 32) parts of one or more drives or three or more (up to 32) entire drives. Data is written to all drives in equal amounts to spread the workload, and parity information is added to the written data to allow for drive failure recovery. This storage scheme provides fault tolerance. If one partition or drive fails in the set, the other members can re-create the missing data on the fly. Once the failed member is replaced or repaired, the data on that drive can be rebuilt and restored. This is also known as disk striping with parity.

Upgrading a drive does not cause data loss or any change in the existing partition structure. Existing drive configurations (mirror, duplex, stripe, and spanned volumes) can be upgraded to dynamic volumes. However, the dynamic disk should be a non-system disk. The drive(s) must have at least 1 MB of unallocated space and you must reboot the computer for the changes to take effect. To upgrade a disk to a dynamic disk:

1. In the Disk Management display, right-click the disk (not volume) you want to upgrade, for example, Disk 1.
2. Choose Upgrade to Dynamic Disk from the menu.

When a drive is converted to dynamic storage, it is labeled as such in Disk Management (see Disk 1 in Figure 4-1).



Volume	Layout	Type	File System	Status
FAT DATA (F:)	Partition	Basic	FAT	Healthy
FAT DATA2 (G:)	Partition	Basic	FAT	Healthy
MS-DOS_6 (C:)	Partition	Basic	FAT	Healthy (
data1-NTFS (…	Partition	Basic	NTFS	Healthy
non disk (J:)	Partition	Basic	NTFS	Healthy
nts-sample (E:)	Partition	Basic	NTFS	Healthy
w2kadv (K:)	Simple	Dynamic	NTFS	Healthy
w2kpro (I:)	Simple	Dynamic	NTFS	Healthy (
w2ksvr (D:)	Simple	Dynamic	NTFS	Healthy

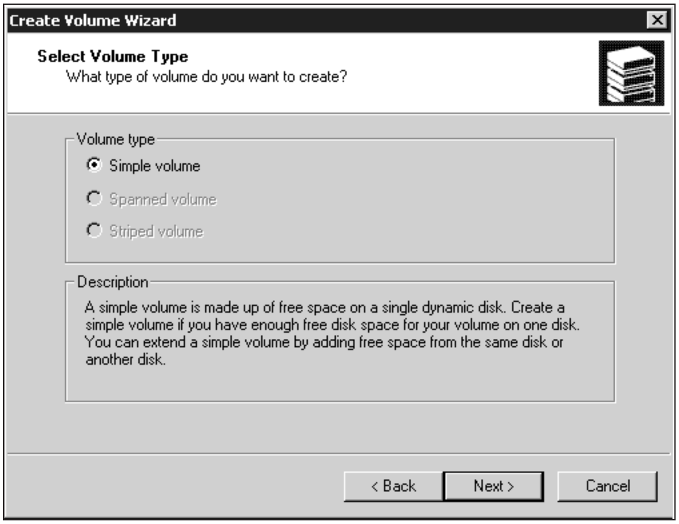
Disk 0 Basic 7.87 GB Online	**MS-D(** 251 ME Health	**nts-samp** 3.71 GB NT Healthy	**FAT DAT** 1.00 GB F Healthy	**FAT DAT** 1.00 GB F Healthy	**data1-N** 1.88 GB N Healthy	**no** 24 He.
Disk 1 Dynamic 8.50 GB Online	**w2kpro (I:)** 1.99 GB NTFS Healthy (Boot)	**w2ksvr (D:)** 2.00 GB NTFS Healthy	**w2kadv (K:)** 2.00 GB NTFS Healthy	2.51 GB Unallocated		

■ Unallocated ■ Primary Partition ■ Extended Partition ■ Logical Drive ■ Simple Volum

Figure 4-1 The Disk Management tool

After you have a dynamic storage host, the next step is to create a volume. To create a volume on a dynamic disk, follow these steps:

1. Right-click over an unallocated dynamic storage device that is not a system disk, and select **Create Volume** from the resulting menu.
2. This launches the Create Volume Wizard. Click **Next**.
3. You'll be prompted as to what type of volume to create (see Figure 4-2). Select one and click **Next**. (See the "Drive Configurations" section later in this chapter.)
4. Now you must select the available dynamic storage devices and how much of each device to use in the volume being created. Click **Next**.
5. Next, you will be prompted to select a drive letter or a mount point, or to not assign a drive letter at all. Click **Next**. (See the "Drive Letters and Mount Points" section later in this chapter.)
6. Finally, you'll be asked whether to format the volume and with what file system. Click **Next**.
7. Click **Finish** to implement volume creation.



Create Volume Wizard

Select Volume Type
What type of volume do you want to create?

Volume type
● Simple volume
○ Spanned volume
○ Striped volume

Description
A simple volume is made up of free space on a single dynamic disk. Create a simple volume if you have enough free disk space for your volume on one disk. You can extend a simple volume by adding free space from the same disk or another disk.

Figure 4-2 Create Volume Wizard, Select Volume Type window

Table 4-1 compares the functions and capabilities of basic and dynamic disks.

Table 4-1 Basic Versus Dynamic Disks

Tasks	Basic Disk	Dynamic Disk
Create and delete primary and extended partitions	X	
Create and delete logical drives within an extended partition	X	
Format and label a partition and mark it active	X	
Delete a volume set	X	
Break a mirror from a mirror set	X	
Repair a mirror set	X	
Repair a stripe set with parity	X	
Upgrade a basic disk to a dynamic disk	X	
Create and delete simple, spanned, striped, mirrored, and RAID 5 volumes	X	X
Extend a volume across one or more disks	X	X
Add a mirror to or remove a mirror from a mirrored volume	X	X
Repair a mirrored volume	X	X
Repair a RAID 5 volume	X	X
Check information about disks, such as capacity, available free space, and current status	X	X
View volume and partition properties such as size	X	X
Make and change drive-letter assignments for hard disk volumes or partitions and CD-ROM devices	X	X
Create volume mount points	X	X
Set or verify disk sharing and access arrangements for a volume or partition	X	X

TIP

Table 4-1 was taken from the *Windows 2000 Server Resource Kit*.

Dynamic drives can be returned to basic storage by deleting all volumes and issuing the Revert To Basic Disk command on the drive through the Disk Management snap-in.

CAUTION

Reverting a disk to basic storage will destroy all data on that drive because you must delete the volumes first.

Removable Storage Devices

The addition of Plug and Play support with Windows 2000 also provides support for **removable storage devices**, such as floppy disks, cartridges, or drives that can be removed between reboots or as a hot swappable device. These removable storage devices or storage media can contain only a single primary partition and cannot participate in dynamic storage. They cannot host extended partitions, nor can they be marked active.

Drive Configurations

Windows 2000 supports several drive configurations. Although Windows 2000 can manage configurations using basic storage partitions, it can create new drive configurations only by using dynamic storage devices. There are five drive configurations or structures supported by Windows 2000: simple volumes, spanned volumes, striped volumes, mirrored volumes, and RAID 5 volumes.

CAUTION

Regardless of what disk configuration you use, always protect your data by using a regularly scheduled backup system.

File Systems

Windows 2000 supports FAT (also known as FAT16), FAT32, and NTFS. FAT is retained by Windows 2000 for backward compatibility with other operating systems. This allows an easy upgrade from another operating system to Windows 2000 and enables multiboot systems to share data drives (when basic storage is used). FAT32 is used to support larger volumes and offers multiboot shared drives with Windows 98 and Windows 95 (OSR2). NTFS is the preferred file system to use with Windows 2000. It offers significantly larger volume support, file-by-file compression, file-by-file security, and more. Windows 2000 NTFS volumes can be accessed by Windows NT 4.0 with Service Pack 4 or higher applied.

TIP

FAT and FAT32 are both referred to as FAT in most Microsoft documentation. The separate terms are used only when the differences between FAT and FAT32 are important.

FAT, FAT32, and NTFS all support **long filenames (LFNs)** with lengths up to 256 characters. FAT and FAT32 store equivalents for DOS-style 8.3 filenames (8 characters plus a 3-character extension) for compatibility with DOS-based utilities that do not recognize LFNs.

TIP

To avoid losing file information, it is important to use utilities that support LFNs when performing any disk or file operation involving LFNs.

FAT and FAT32

FAT was originally developed for DOS. It has experienced several revisions and improvements as support for FAT was included in newer operating systems. FAT under Windows 2000 maintains backward compatibility with previous operating systems (DOS, Windows 3.1x, and Windows for Workgroups) while supporting newer features or capabilities. In addition, FAT is most often used to format floppies and other removable media in Windows 2000.

The important features of FAT (under Windows 2000) are:

- Supports volumes up to 4 GB in size
- Most efficient on volumes smaller than 256 MB
- Root directory can contain only 512 entries
- No file-level compression
- No file-level security

FAT32 is simply an enhanced version of FAT that was originally released with Windows 95 OSR2. FAT32's main feature change from FAT is that of volume size. Windows 2000 can support and access FAT32 volumes up to 2 TB in size, but only volumes up to 32 GB can be created. FAT32 volumes have a minimum size of 512 MB.

A FAT volume is divided into clusters. A **cluster** is one or more sectors grouped into a single nondivisible unit. If cluster size is not specified, Windows 2000 configuration will use the default, which varies according to disk size to reduce the amount of space lost and fragmentation in the volume. A **sector** is the smallest division (512 bytes) of a drive's surface. Because of the limitations of the file system, only a maximum number of clusters can be addressed. For FAT16, the maximum number of clusters is 65,536. For FAT32, the maximum number of clusters is 268,435,456 (see Table 4-2).

Table 4-2 FAT16 and FAT32 Cluster Sizes

Drive Size	FAT16 Cluster Size	FAT32 Cluster Size
260 MB–511 MB	8 KB	4 KB
512 MB–1,023 MB	16 KB	4 KB
1,024 MB–2 GB	32 KB	4 KB
2 GB–4 GB	64 KB	4 KB
4 GB–8 GB	Not supported	4 KB
8 GB–16 GB	Not supported	8 KB
16 GB–32 GB	Not supported	16 KB
<\>>32 GB	Not supported	Not supported

Before Windows 95, the maximum volume size of FAT was 2 GB. With the use of 64 KB clusters, this was extended to 4 GB. However, 64 KB clusters can cause problems with some drive utilities. Thus, Windows 2000 will always warn you when you attempt to format a 2 GB to 4 GB partition with FAT16.

NTFS

NTFS is the preferred file system of Windows 2000. The important features of NTFS are:

- Supports volumes up to 2 TB in size (larger sizes are possible, but not recommended by Microsoft)
- Most efficient on volumes larger than 512 MB
- Root directory can contain unlimited entries
- File-level compression
- File-level security
- File-level encryption
- Disk quotas (a **disk quota** is a limitation on the amount of disk space that can be consumed by a user)
- POSIX support

The version of NTFS included with Windows 2000 is different from that of Windows NT 4.0. In fact, you must have Service Pack 4 or higher installed on Windows NT 4.0 to access Windows 2000 NTFS volumes. Microsoft does not recommend a multiboot system with Windows NT and Windows 2000 for this reason.

TIP

FAT and FAT32 volumes on a system can be migrated to the Windows 2000 NTFS format without losing data. However, to return to FAT, the volume must be deleted, re-created, formatted, and the data copied back onto the new volume.

NTFS manages clusters more efficiently than FAT32 (see Table 4-3).

Table 4-3 NTFS Default Cluster Sizes

Volume Size	Sectors Per Cluster	Cluster Size
512 MB or less	1	512 bytes
513 MB–1,024 MB	2	1 KB
1,025 MB–2,048 MB	4	2 KB
2,049 MB–4,096 MB	8	4 KB
4,097 MB–8,192 MB	16	4 KB
8,193 MB–16,384 MB	32	4 KB
16,385 MB–32,768 MB	64	4 KB
> 32,768 MB	128	4 KB

TIP

File-level compression cannot be used on volumes with a cluster size greater than 4 KB.

POSIX Support via NTFS

Windows 2000 comes with various environmental subsystems. The POSIX subsystem is designed to run POSIX applications and meets the requirements of the POSIX.1 government standard.

POSIX (Portable Operating System Interface for Computing Environments) is a set of standards drafted by the Institute of Electrical and Electronic Engineers (IEEE) that defines various aspects of an operating system, and includes topics such as programming interface, security, networking, and graphical interface. So far, only one of these standards, POSIX.1, has made the transition from draft to final form. It's not widely used, but sufficiently so that POSIX compatibility was necessary for Windows 2000 to be acceptable to the U.S. Department of Defense.

POSIX.1 is based on ideas drawn from the UNIX file system and process model. Because POSIX.1 addresses only API (application programming interface) issues, most applications written to the POSIX.1 API must rely on non-POSIX operating system extensions (in this case, Win32) to provide services such as security and networking.

POSIX applications need certain file system functionality, such as support for case-sensitive filenames (in POSIX, there's a difference between MyFile.doc, MYFILE.DOC, and myfile.doc) and support for files with multiple names (or hard links). NTFS supports these

POSIX requirements. Any POSIX application that requires access to file system resources must have access to an NTFS partition, but POSIX applications that do not access file system resources can run on FAT

CAUTION

If you install POSIX utilities or file systems on your Windows 2000 system, be sure to use native POSIX file management utilities to manage them. Just as older DOS utilities will destroy LFN information created by FAT or FAT32, native Windows 2000 file utilities—Windows Explorer, File Manager, and My Computer—will destroy POSIX file structures, especially when the only difference between two or more otherwise identical filenames is their use of uppercase and lowercase characters. For example, POSIX understands very well that MyFile.txt is different from myfile.txt, but NTFS does not (and will actually destroy the folder entry for whichever name appears second in the POSIX-created folder structure).

File Compression

File compression is the ability to compress data on the basis of single files, folders, or entire volumes. File compression offers the benefit of being able to store more data in the same space, but at the cost of some performance. The amount of compression achieved depends on the data stored in the object (that is, text can often be compressed significantly, whereas executable programs can not). Windows 2000 manages compression via the NTFS file system drive. Each time a compressed file is read, it must be uncompressed as it is read. Likewise, saving a compressed file, copying a file into a compressed folder, or creating a new file in a compressed folder requires that the data to be stored is compressed in memory before it is written to the drive.

Configuring and managing file compression involves enabling or disabling the file compression attribute on one or more files or folders. File compression appears as one of the attributes of NTFS file/folder objects on the Advanced Attributes dialog box (see the section entitled "NTFS File Object" later in this chapter; try Hands-on Project 4-11). And, just like all other attributes, file compression can be set on a file-by-file basis or by setting the attribute on a container. When the "Compress contents to save disk space" checkbox is selected, the object(s) are compressed. When this checkbox is cleared, the object(s) are expanded back to their original size.

TIP

Troubleshooting file compression usually involves either recompressing or removing compression from files, or restoring files from backup that were damaged while they were compressed.

Converting Between File Systems

When you first format a drive under Windows 2000, you have the option of selecting FAT, FAT32, or NTFS. If at a later date you decide you need to change the file system, you have only two options: reformat with the new file system or convert from FAT/FAT32 to NTFS.

CAUTION

Before any file system conversion, be sure to back up the drive to ensure you will not lose data.

To reformat, simply employ one of the disk tools, such as Disk Management, and format the volume with a new file system. Remember that all data stored on the drive will be lost, so without a backup you will not be able to recover from a format.To convert from FAT/FAT32 to NTFS, you use the Convert.exe command-line tool (try Hands-on Project 4-14). It has two parameters: /fs:ntfs and /v.The first specifies the conversion should result in the NTFS file system (yes, it is strange to have this parameter because it only supports conversion to NTFS). The second turns on "verbose" mode so all messages regarding the conversion are displayed. When launched, CONVERT will attempt to convert the drive immediately. If the drive is locked (i.e., a process has an open file from the volume to be converted), the conversion will occur during the next bootup of the system.

DISK MANAGEMENT ACTIONS

In addition to creating volumes and transforming devices into dynamic storage, the Disk Management tool offers several other useful features.The All Tasks submenu of the Action menu is context-based, depending on what type of object is selected.The All Tasks submenu is the same menu that pops up when you right-click a drive, partition, or volume object.The commands that appear in this menu are:

- *Add Mirror*—Adds a mirror to duplicate a partition or volume (not available in Windows 2000 Professional)
- *Change Drive Letter and Path*—Changes the drive letter of basic disks and dynamic disks or the mount path of dynamic disks
- *Create Partition*—Creates a partition on a basic disk
- *Create Volume*—Creates a volume on a dynamic disk
- *Delete Partition*—Destroys a partition, returning the space to unallocated status
- *Explore*—Opens the selected volume or partition into a Windows Explorer window
- *Extend Volume*—Adds additional unallocated space to an existing volume
- *Format*—Formats a volume or partition with a file system
- *Help*—Opens the help utility
- *Import Foreign Disks*—Imports a dynamic disk when moved from one Windows 2000 computer to another
- *Mark Partition Active*—Marks a primary partition active
- *Open*—Opens the selected volume or partition into a My Computer window
- *Properties*—Opens the Properties dialog box for the selected object

- *Reactivate Disk*—Brings dynamic disks back online after being powered down, disconnected, or corrupted.
- *Reactivate Volume*—Recovers volumes from a failed status
- *Remove Disk*—Used to deactivate a removable drive
- *Revert to Basic Disk*—Transforms a dynamic disk into a basic disk; requires that all volumes be deleted
- *Upgrade to Dynamic Disk*—Transforms a basic storage device into a dynamic storage device

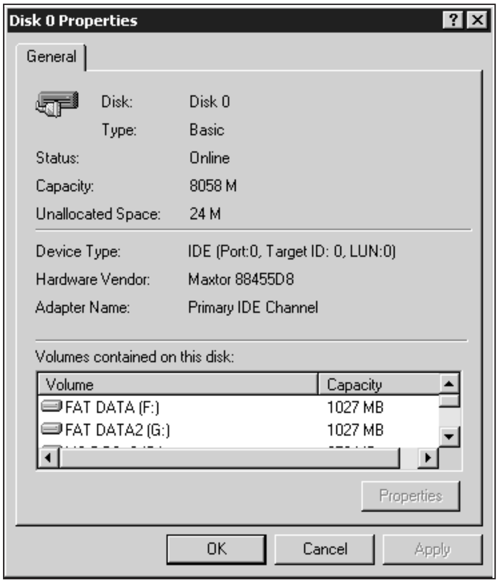
The Action menu itself has three other non-context-sensitive commands:

- *Refresh*—Updates drive letters, file system, volume, and removable media information and determines which previously unreadable volumes are now readable
- *Rescan Disks*—Updates hardware information by rescanning all attached storage devices (including removable media) for changes in configuration
- *Restore Basic Disk Configuration*—Using a disk configuration file saved from Disk Administrator from Windows NT (4.0 or earlier), restores the state of the physical hard drives to their basic configurations under Windows NT

> TIP
>
> Disk Management can be used to manipulate storage devices on remote computers. Simply select the "Computer Management (local)" item in the console tree and issue the "Connect to another computer" command from the Action menu. This opens a list of all known networked systems. Once you've selected another system, you can perform disk management functions as if you were present locally.

The Properties dialog boxes of drives, volumes, and partitions offer lots of details and configuration settings. A drive (disk, not volume or partition) Properties dialog box (see Figure 4-3) will display the following information:

- *Disk*—The ordinal number of the disk, such as Disk 0, Disk 1, etc.
- *Type*—The storage type: basic, dynamic, or removable
- *Status*—The status of the device: online, offline, foreign, or unknown
- *Capacity*—The maximum storage capacity of the drive
- *Unallocated Space*—The amount of space not used in a partition or volume
- *Device Type*—IDE, EIDE, or SCSI, plus type-specific details
- *Hardware Vendor*—Hardware vendor name and disk model
- *Adapter Name*—Type of drive controller to which the drive is attached
- *Volumes contained on this disk*—The volumes and capacity of each volume or partition on the drive



Disk: Disk 0
Type: Basic
Status: Online
Capacity: 8058 M
Unallocated Space: 24 M

Device Type: IDE (Port:0, Target ID: 0, LUN:0)
Hardware Vendor: Maxtor 88455D8
Adapter Name: Primary IDE Channel

Volumes contained on this disk:

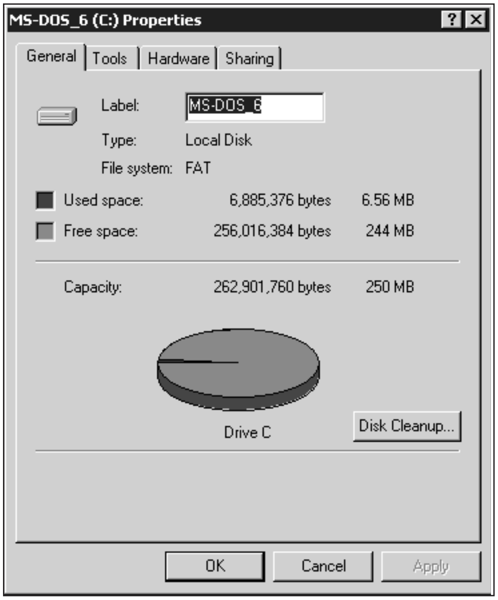
Volume	Capacity
FAT DATA (F:)	1027 MB
FAT DATA2 (G:)	1027 MB

Figure 4-3 A drive Properties dialog box, General tab

The Properties dialog box for a partition or a volume displays the same detailed information. However, an NTFS-formatted partition or volume has two additional tabs that are not present on FAT/FAT32 formatted partitions or volumes. The tabs of the Properties dialog box are: General, Tools, Hardware, Sharing, Security, and Quota (the latter two are NTFS only).

The General tab (see Figure 4-4) displays:

- *Label*—The customizable name of the disk. FAT and FAT 32 drives can be labeled with up to 11 characters, whereas NTFS labels can contain 32 characters.
- *Type*—The type of disk: local, network connection, floppy disk drive, CD-ROM drive, RAM disk, removable drive, or mounted disk
- *File System*—The file system used on the disk: CDFS (for CDs); FAT, FAT32, NTFS, or UDF (Universal Disk Format is common on DVD and compact-discs)
- *Used Space*—The amount of space used by stored files
- *Free Space*—The amount of space still available in the partition
- *Capacity*—The total amount of space in the partition
- *Graph*—A graphical pie chart representation of used and free space
- *Disk Cleanup*—A button to access the Disk Cleanup tool (discussed later in this chapter)



Label: MS-DOS_6

Type: Local Disk

File system: FAT

■ Used space:	6,885,376 bytes	6.56 MB
▨ Free space:	256,016,384 bytes	244 MB
Capacity:	262,901,760 bytes	250 MB

Figure 4-4 A Partition or Volume Properties dialog box, General tab

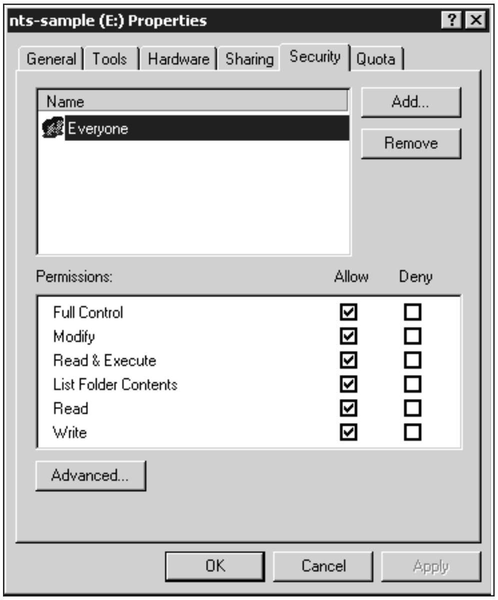
The Tools tab offers access to:

- *Error-checking:* Accesses the ScanDisk tool to find and repair errors on a drive (discussed later in this chapter)
- *Backup:* Accesses the NT Backup utility to back up files
- *Defragmentation:* Accesses the Defragmentation tool to reduce file fragmentation (discussed later in this chapter)

The Hardware tab lists all physical storage devices and their type. This dialog box accesses the same Troubleshooting and Properties (for drivers) utilities as those accessed through the Device Manager.

The Sharing tab is used to share partitions with the network (discussed later in this chapter).

The Security tab (see Figure 4-5) is used to set the NTFS access permissions on the volume or partition as a whole. Individual users or groups each can be defined with unique permissions of allow or deny for each of the listed object-specific actions (discussed later in this chapter).



Name

Everyone

Permissions:	Allow	Deny
Full Control	☑	□
Modify	☑	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	☑	□

Figure 4-5 An NTFS Partition or Volume Properties dialog box, Security tab

The Quota tab is used to define disk use limitations on NTFS volumes and partitions. The quota can be defined on a general basis and/or fine-tuned for each individual user. Quota tab options include:

- *Enable quota management*—Turns on the quota system
- *Deny disk space to users exceeding quota limit*—Prevents users from gaining more space when in violation of the quota
- *Do not limit disk usage*—Disables system-wide quota level
- *Limit disk space to*—Sets the maximum amount of drive space that can be accessed by a single user
- *Set warning level to*—Sets a threshold that when crossed warns the user about nearing their quota limit
- *Log event when a user exceeds their quota limit*—Adds an item to the Event Viewer (an application that displays event logs)
- *Log event when a user exceeds their warning level*—Adds an item to the Event Viewer
- *Quota Entries*—Opens a dialog box where quota settings for each user can be fine-tuned

Assigning Drive Letters and Mount Points

Windows 2000 uses two methods to access formatted volumes—drive letters and mount points. **Drive letters** are used to grant applications and user interface utilities access to file system resources. Drive letters A and B are typically used for floppies, but in the absence of floppies these letters can be employed as mappings for network shares. (A **share** is a resource—such as an application, file, or printer—that can be accessed or shared over the network.) Drive letters C through Z are used for local hard drives or mappings for network shares. Even without floppies, the first hard drive is always labeled with C. The drive letters of the system and boot partitions/volumes cannot be changed, but all other drive letters can be changed. The "Change Drive Letter and Path" command in the Disk Management console is used to alter a drive letter, apply a mount point path, or remove a drive letter.

A **mount point** or **mounted volume** is an alternative to drive letters. A mount point connects a FAT/FAT32 or NTFS volume or partition to an empty directory on an NTFS volume or partition. This allows more than 24 (or 26, up to 32) hard drives to be present on a single machine. The empty directory becomes the gateway to the linked volume. A mount point is created by following this procedure:

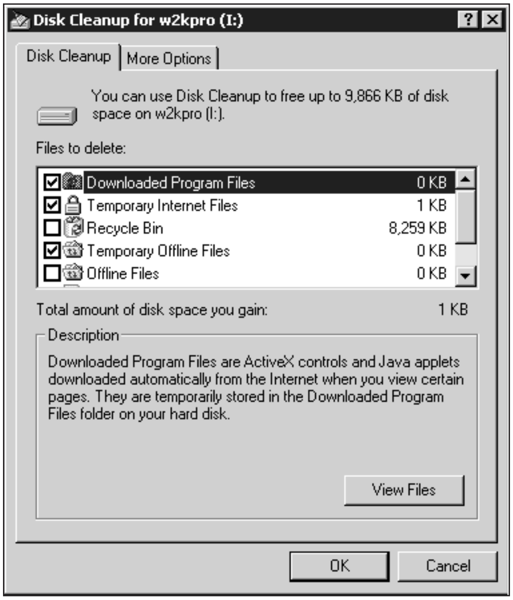
1. Create an empty directory.
2. Open the **Disk Management** tool (**Start**, **Programs**, **Administrative Tools**, **Computer Management**; **Storage**; **Disk Management**).
3. Right-click the volume or partition to be mapped, and then select **Change Drive Letter and Path** from the resulting menu.
4. Click **Add**.
5. Select **Mount in this NTFS folder**.
6. Click **Browse**.
7. Locate and select the empty folder, and then click **OK**.
8. Click **OK**.

TIP It is possible to create an infinite regression mount point (a pointer that loops back on itself) by mapping a volume to an empty directory that it hosts. Although this is a valid procedure, it can cause system overflows when disk utilities attempt to follow the infinite path.

Freeing Disk Space

Disk Cleanup is a tool used to free up space on hard drives by removing deleted, orphaned, temporary, or downloaded files. This utility can be launched from the General tab of the Properties dialog box from any hard drive, or via Start, Programs, Accessories, System Tools, Disk Cleanup. When launched from a drive's properties dialog box, Disk Cleanup will automatically scan that drive for space that can be freed. When launched from the Start menu, you are prompted to select the hard drive to scan for cleaning. The scanning process can take several minutes, especially on large hard drives with a significant number of files.

When scanning is complete, the Disk Cleanup for (drive:) dialog box is displayed (see Figure 4-6). The Disk Cleanup tab of this dialog box lists the file types that can be removed and how much space they currently use. The View Files button can be used to see the selected file type's details via a My Computer window. Selecting the check box beside a listed file type will cause those files to be deleted (not placed in the Recycle Bin) when you click OK.



You can use Disk Cleanup to free up to 9,866 KB of disk space on w2kpro (I:).

Files to delete:

☑ Downloaded Program Files	0 KB
☑ Temporary Internet Files	1 KB
□ Recycle Bin	8,259 KB
☑ Temporary Offline Files	0 KB
□ Offline Files	0 KB

Total amount of disk space you gain: 1 KB

Description

Downloaded Program Files are ActiveX controls and Java applets downloaded automatically from the Internet when you view certain pages. They are temporarily stored in the Downloaded Program Files folder on your hard disk.

Figure 4-6 Disk Cleanup tab

The More Options tab offers access to the Add/Remove Windows Components utility and the Change or Remove Programs utility. These are the same utilities as those accessed through the Add/Remove Programs applet in the Control Panel.

Using ScanDisk

ScanDisk is a disk integrity inspection utility, accessed from the Start menu, Programs, Accessories, System Tools, ScanDisk. It is used to locate both logical and physical errors on a hard drive. Physical errors are marked and are avoided in all future drive accesses by the operating system. Logical errors are bad pointers in the directory structure of a file system, whether FAT, FAT32, or NTFS and often can be corrected. However, when correction is not possible, ScanDisk saves the data of orphaned fragments to text files in the root directory of the drive and uses incremental filenames of FILE0001, FILE0002, etc.

Once launched, it prompts you whether to "automatically fix file system errors" and whether to "scan for and attempt recovery of bad sectors." ScanDisk usually requires rebooting the system before it scans NTFS volumes so that no files are in use when it performs its check.

Windows 2000 automatically starts ScanDisk when it detects an improper system shutdown or errors in the directory structure of a drive. This usually occurs during booting and the execution process. Results are displayed on the blue screen where the operating system name, version, and build, along with processors and memory size, are detailed.

CAUTION

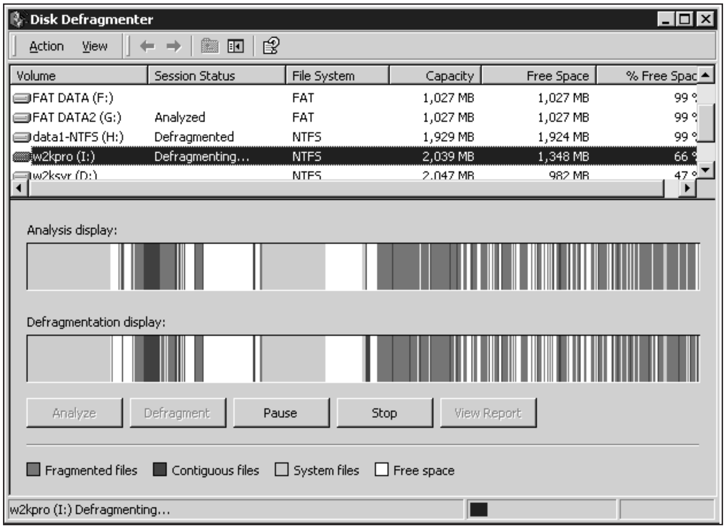
> The ScanDisk that ships with Windows 2000 is specifically designed to manage the file systems supported by Windows 2000. Do not use ScanDisk from any other operating system to attempt repairs on Windows 2000 hard drives.

Defragmenting Hard Drives

As files are written, altered, deleted, rewritten, etc., the storage device develops gaps between used and unused space. When gaps are used instead of contiguous free space to store files, fragmentation occurs. **Fragmentation** is the division of a file into two or more parts where each part is stored in a different location on the hard drive. As the level of fragmentation on a drive increases, it takes longer for read and write operations to occur. **Defragmentation** is the process of reorganizing files so they are stored contiguously and no gaps are left between files.

The Windows 2000 defragmentation utility is designed for FAT, FAT32, and NTFS volumes. It is accessed either from the Tools tab of a drive's Properties dialog box or via Start, Programs, Accessories, System Tools, Disk Defragmenter.

The Disk Defragmenter (see Figure 4-7) lists all drives in the system. When you select a drive, you can either Analyze the drive for fragmentation or go ahead and defragment the drive. Both processes display a graphical representation of the file storage condition of the drive. When either process is complete, you can view a report that details the findings of the procedure.



Volume	Session Status	File System	Capacity	Free Space	% Free Spac
FAT DATA (F:)		FAT	1,027 MB	1,027 MB	99 %
FAT DATA2 (G:)	Analyzed	FAT	1,027 MB	1,027 MB	99 %
data1-NTFS (H:)	Defragmented	NTFS	1,929 MB	1,924 MB	99 %
w2kpro (I:)	Defragmenting...	NTFS	2,039 MB	1,348 MB	66 %
w2ksvr (D:)		NTFS	2,047 MB	982 MB	47 %

▨ Fragmented files ■ Contiguous files □ System files □ Free space

w2kpro (I:) Defragmenting...

Figure 4-7 Disk Defragmenter

The Disk Defragmenter does not offer a built-in scheduling feature, nor can it be executed from a command line. Thus, you must defragment manually or deploy a third-party utility that automates scheduled defragmentation.

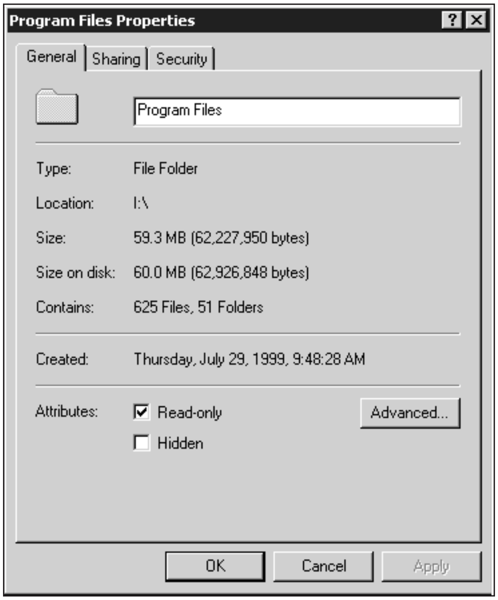
FILE SYSTEM OBJECT-LEVEL CONTROLS

In addition to the drive and volume/partition level controls for storage devices, there are file system controls for folders, files, and mounted volumes. Because folders, files, and mounted volumes are considered objects in Windows 2000, these controls are called object-level controls, and are accessed via the Properties dialog boxes of either a folder or an object. There are minor differences in the dialog boxes depending on whether the file system is FAT/FAT32 or NTFS. There is no difference in file system objects due to having basic or dynamic disks as hosts.

The following sections detail the differences in Properties dialog boxes for each object type. The Sharing and Security tabs of these dialog boxes are discussed in a later section in this chapter.

NTFS Folder Object

An NTFS folder object's Properties dialog box has three tabs: General (see Figure 4-8), Sharing, and Security.



Program Files Properties

General | Sharing | Security

Program Files

Type: File Folder

Location: I:\

Size: 59.3 MB (62,227,950 bytes)

Size on disk: 60.0 MB (62,926,848 bytes)

Contains: 625 Files, 51 Folders

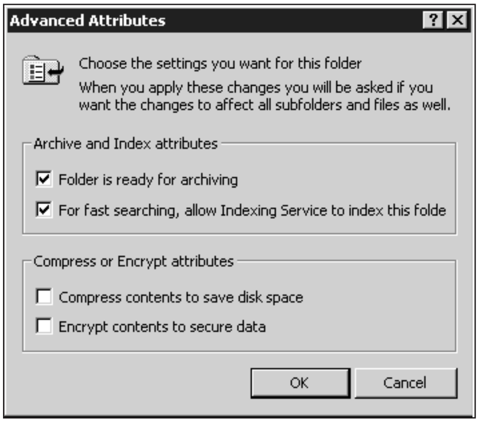
Created: Thursday, July 29, 1999, 9:48:28 AM

Attributes: ☑ Read-only ☐ Hidden

Figure 4-8 An NTFS folder object Properties dialog box, General tab

The General tab offers the following information:

- *Name*—The customizable name of the object
- *Type*—Lists object type: File Folder
- *Location*—The path of the object
- *Size*—The byte size of the object, including its contents
- *Size on disk*—The actual amount of drive space used to store the object
- *Contains*—Lists the number of files and folders the object contains
- *Created*—Lists the object's creation date and time
- *Attributes: Read-only*—A check box used to prevent writing to, changing, or deleting the object
- *Attributes: Hidden*—A check box used to hide the object from view
- *Advanced button: Folder is ready for archiving*—A check box that indicates that this folder, and optionally its contents, is ready for backup (see Figure 4-9)
- *Advanced button: For fast searching, allow Indexing Service to index this folder*—A check box that when selected preindexes the folder, and optionally its contents, for faster searching
- *Advanced button: Compress contents to save disk space*—A check box used to compress the folder, and optionally its contents
- *Advanced button: Encrypt contents to secure data*—A check box used to encrypt the folder, and optionally its contents



Advanced Attributes

Choose the settings you want for this folder

When you apply these changes you will be asked if you want the changes to affect all subfolders and files as well.

Archive and Index attributes

☑ Folder is ready for archiving

☑ For fast searching, allow Indexing Service to index this folde

Compress or Encrypt attributes

□ Compress contents to save disk space

□ Encrypt contents to secure data

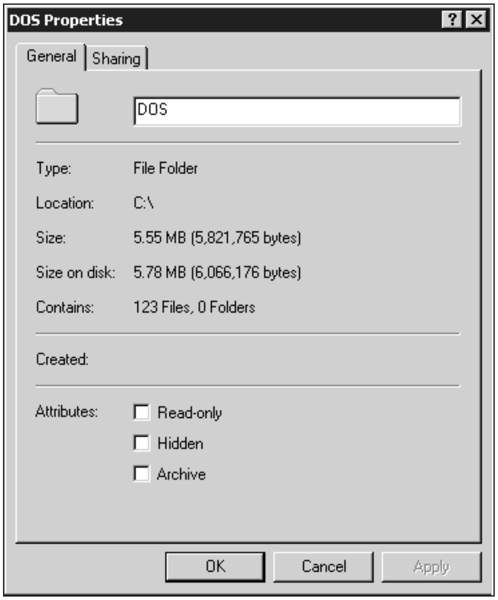
Figure 4-9 The Advanced Attributes dialog box of an NTFS object

When the Properties dialog box for the object is closed, all changes to the settings via the Advanced button will require confirmation by clicking the OK button.

FAT/FAT32 Folder Object

A FAT/FAT32 folder object's Properties dialog box has two tabs: General (see Figure 4-10) and Sharing. The General tab offers the following information:

- *Name*—The customizable name of the object
- *Type*—Lists object type: File Folder
- *Location*—The path of the object
- *Size*—The byte size of the object, including its contents
- *Size on disk*—The actual amount of drive space used to store the object
- *Contains*—Lists the number of files and folders it contains
- *Created*—Lists the object's creation date and time
- *Attributes: Read-only*—A check box used to prevent writing to, changing, or deleting the object
- *Attributes: Hidden*—A check box used to hide the object from view
- *Attributes: Archive*—A check box that indicates that this object should be included in the next backup operation



DOS

Type: File Folder

Location: C:\

Size: 5.55 MB (5,821,765 bytes)

Size on disk: 5.78 MB (6,066,176 bytes)

Contains: 123 Files, 0 Folders

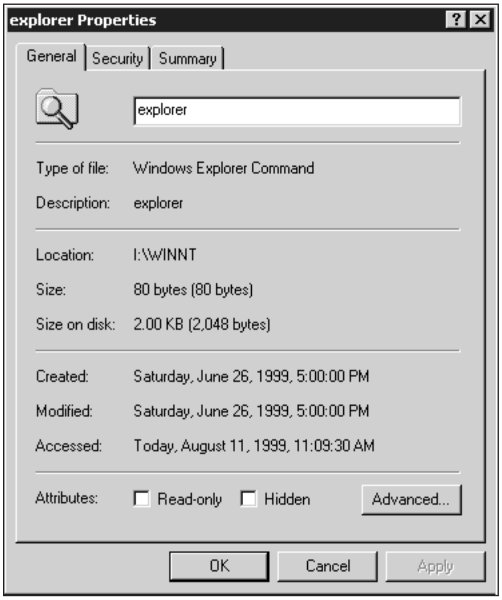
Created:

Attributes: ☐ Read-only ☐ Hidden ☐ Archive

Figure 4-10 A FAT/FAT32 folder object Properties dialog box, General tab

NTFS File Object

An NTFS file object's Properties dialog box has three tabs: General (see Figure 4-11), Security, and Summary.



explorer

Type of file: Windows Explorer Command

Description: explorer

Location: I:\WINNT

Size: 80 bytes (80 bytes)

Size on disk: 2.00 KB (2,048 bytes)

Created: Saturday, June 26, 1999, 5:00:00 PM

Modified: Saturday, June 26, 1999, 5:00:00 PM

Accessed: Today, August 11, 1999, 11:09:30 AM

Attributes: □ Read-only □ Hidden

Figure 4-11 An NTFS file object Properties dialog box, General tab

The General tab offers the following information:

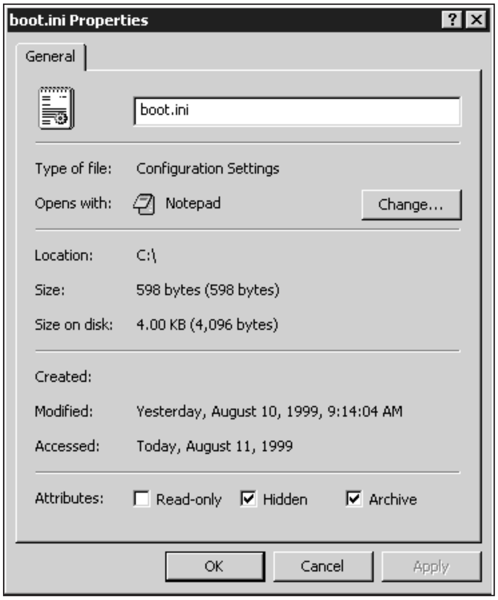
- *Name*—The customizable name of the object
- *Type of File*—Names the file type or defines it as a *blank* file where *blank* is the file's extension
- *Description (application files only)*—Names the utility or application
- *Opens with (nonapplication files only)*—Lists the application used to open the file
- *Change (nonapplication files only)*—A button to alter the application used to open the file
- *Location*—The path of the object
- *Size*—The byte size of the object
- *Size on disk*—The actual amount of drive space used to store the object
- *Created*—Lists the creation time and date of the object
- *Modified*—Lists the last time and date of a change to the object
- *Accessed*—Lists the last time and date the object was accessed
- *Attributes: Read-only*—A check box used to prevent writing to, changing, or deleting the object

- *Attributes: Hidden*—A check box used to hide the object from view
- *Advanced button: File is ready for archiving*—A check box that indicates that this file is ready for backup
- *Advanced button: For fast searching, allow Indexing Service to index this file*—A check box that when selected preindexes the object for faster searching
- *Advanced button: Compress contents to save disk space*—A check box used to compress the object
- *Advanced button: Encrypt contents to secure data*—A check box used to encrypt the object

The Summary tab is used to define description and origin details for the object. These details include title, subject, category, keywords, comments, source, author, and revision number. This information can be used to refine searches.

FAT/FAT32 File Object

A FAT/FAT32 file object's Properties dialog box has only the General tab (see Figure 4-12).



boot.ini

Type of file: Configuration Settings

Opens with: Notepad

Location: C:\

Size: 598 bytes (598 bytes)

Size on disk: 4.00 KB (4,096 bytes)

Created:

Modified: Yesterday, August 10, 1999, 9:14:04 AM

Accessed: Today, August 11, 1999

Attributes: □ Read-only ☑ Hidden ☑ Archive

Figure 4-12 A FAT/FAT32 file object Properties dialog box, General tab

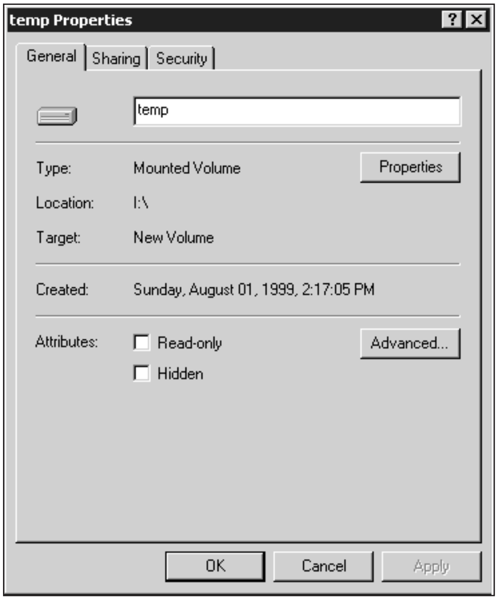
The General tab offers the following information:

- *Name*—The customizable name of the object
- *Type of File*—Names the file type or defines it as a *blank* file where *blank* is the file's extension
- *Description (application files only)*—Names the utility or application

- *Opens with (nonapplication files only)*—Lists the application used to open the file
- *Change (nonapplication files only)*—A button for altering the application used to open the file
- *Location*—The path of the object
- *Size*—The byte size of the object
- *Size on disk*—The actual amount of drive space used to store the object
- *Created*—Lists the object's creation date and time
- *Modified*—Lists the last time and date of a change to the object
- *Accessed*—Lists the last time and date the object was accessed
- *Attributes: Read-only*—A check box used to prevent writing to, changing, or deleting the object
- *Attributes: Hidden*—A check box used to hide the object from view
- *Attributes: Archive*—A check box that indicates that this object should be included in the next backup operation

NTFS Mounted Volume Object

An NTFS mounted volume object's Properties dialog box has three tabs: General (see Figure 4-13), Sharing, and Security.


temp Properties

General | Sharing | Security

temp

Type: Mounted Volume

Location: I:\

Target: New Volume

Created: Sunday, August 01, 1999, 2:17:05 PM

Attributes: □ Read-only □ Hidden

Figure 4-13 An NTFS mounted volume object Properties dialog box, General tab

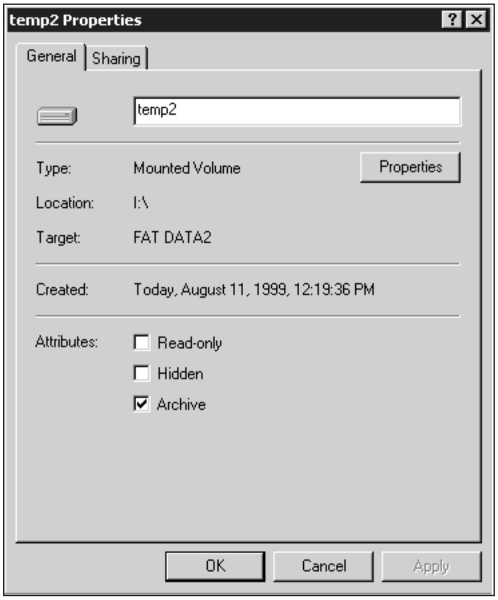
The General tab offers the following information:

- *Name*—The customizable name of the object
- *Properties*—A button used to access the mounted volume's Properties dialog box, the same dialog box that would be seen via Disk Management
- *Type*—Lists object type: Mounted Volume
- *Location*—The path of the object
- *Target*—Names the mapped volume
- *Created*—Lists the object's creation date and time
- *Attributes: Read-only*—A check box used to prevent writing to, changing, or deleting the object
- *Attributes: Hidden*—A check box used to hide the object from view
- *Advanced button: Folder is ready for archiving*—A check box that indicates that this folder, and optionally its contents, is ready for backup
- *Advanced button: For fast searching, allow Indexing Service to index this folder*—A check box that, when selected, preindexes the folder, and optionally its contents, for faster searching
- *Advanced button: Compress contents to save disk space*—A check box used to compress the folder, and optionally its contents
- *Advanced button: Encrypt contents to secure data*—A check box used to encrypt the folder, and optionally its contents

FAT/FAT32 Mounted Volume Object

A FAT/FAT32 mounted volume object's Properties dialog box has two tabs: General (see Figure 4-14) and Sharing. The General tab offers the following information:

- *Name*—The customizable name of the object
- *Properties*—A button used to access the mounted volume's Properties dialog box, the same dialog box that would be seen via Disk Management
- *Type*—Lists object type: Mounted Volume
- *Location*—The path of the object
- *Target*—Names the mapped volume
- *Created*—Lists the object's creation date and time
- *Attributes: Read-only*—A check box used to prevent writing to, changing, or deleting the object
- *Attributes: Hidden*—A check box used to hide the object from view
- *Attributes: Archive*—A check box that indicates that this object should be included in the next backup operation



temp2

Type: Mounted Volume

Location: I:\

Target: FAT DATA2

Created: Today, August 11, 1999, 12:19:36 PM

Attributes: □ Read-only □ Hidden ☑ Archive

Figure 4-14 A FAT/FAT32 mounted volume object Properties dialog box, General tab

MANAGING NTFS PERMISSIONS

The NTFS file system offers file-level control over access on a user and group basis. The only file system supported by Windows 2000 that offers file level security is NTFS. NTFS security determines what can be done to a file system object and who can perform those actions. There are different permissions for folders and files.

NTFS File and Folder Permissions

NTFS file and folder permissions are nearly identical. The dialog boxes and control interfaces for files and folders are the same. The only differences are: files do not offer child inheritance options (because files are child objects, they do not have child objects themselves) and some obvious permissions apply only to folders or only to files.

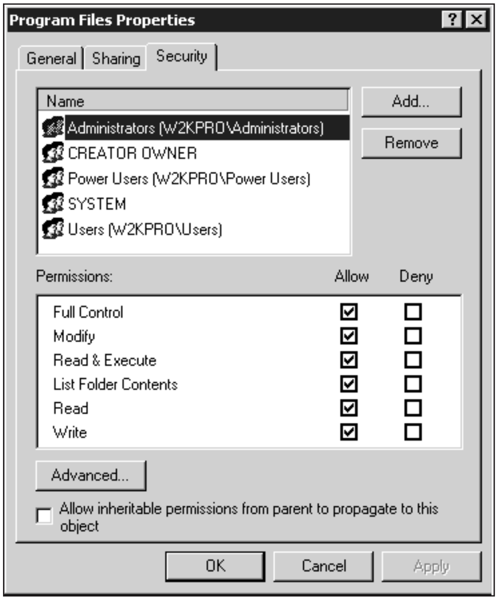
> TIP
>
> Inheritance is the mechanism by which the contents of a container receive the same settings as the container. Inheritance allows administrators to change a setting on a folder, drive, group, policy, etc., and then force those changes to be applied to all the objects contained within the folder, drive, group, etc.

In some cases, the same permission name has a different meaning for files and folders. In other cases, similar permissions have different names but both names are listed in both dialog box contexts. The NTFS permissions are as follows:

- *Read*—Allows users to view and access the contents of the folder or the file

- *Write (folders)*—Allows users to create new folders and files within the folder
- *Write (files)*—Allows users to overwrite the file and change attributes
- *List Folder Contents (folders only)*—Allows users to see the names of the contents of the folder
- *Read & Execute (folders)*—Allows users to reach files and folders via folders where they do not have access permission; also allows users to view and access the contents of the folder
- *Read & Execute (files)*—Allows users to run applications and to view and access the file
- *Modify (folders)*—Allows users to delete the folder and its contents; also allows users to create new folders and files within the folder and to view and access the contents of the folder
- *Modify (files)*—Allows users to delete the file, to overwrite the file, and change attributes, to run applications and view and access the file
- *Full Control (folders)*—Grants users complete and unrestricted access to all functions of the folder and its contents
- *Full Control (files)*—Grants users complete and unrestricted access to all functions of the file

The NTFS permissions are configured on the Security tab of the object's Properties dialog box (see Figure 4-15). The controls this tab offers are discussed in the following section.



Name
- Administrators (W2KPRO\Administrators)
- CREATOR OWNER
- Power Users (W2KPRO\Power Users)
- SYSTEM
- Users (W2KPRO\Users)

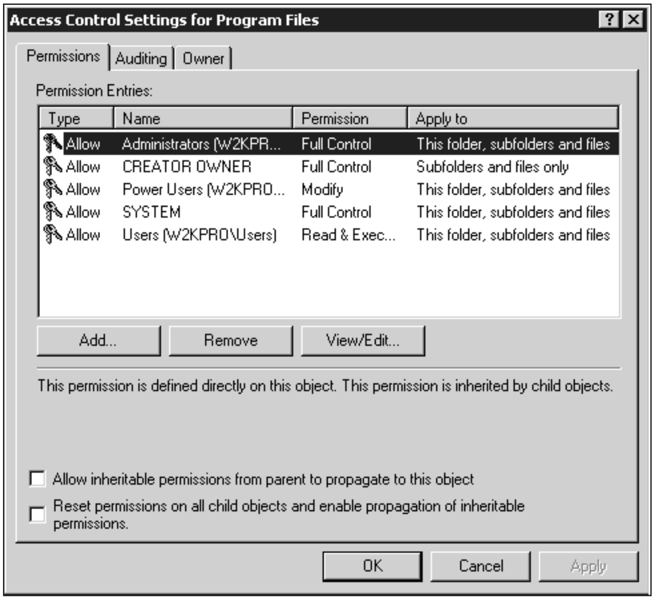
Permissions:	Allow	Deny
Full Control	☑	□
Modify	☑	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	☑	□

□ Allow inheritable permissions from parent to propagate to this object

Figure 4-15 An NTFS folder object Properties dialog box, Security tab

To change permissions for a user or group, select that user or group in the Name list. If the user or group is not present, use the Add button to include that user or group in the list. After a user or group is selected, the Permissions field displays the current settings for that specific selection. Selecting or deselecting the Allow or Deny check boxes for each permission level defines the custom permissions for the selected user or group. To remove a user or group, select it in the Name list and click the Remove button. When a user or group is not listed on the Security tab for an object, that user or group has no effective permissions to that object. In other words, the user or group is prevented from accessing the object.

Clicking the Advanced button reveals a three-tabbed dialog box where more detailed access control settings can be defined. The Permissions tab (see Figure 4-16) of the Access Control Settings dialog box is used to define detailed permissions on a per user or per group basis. Similarly to the previous dialog box, users and groups are included in the list via the Add button and deleted with the Remove button. This dialog box also offers two more check boxes. The first check box is the same inheritable permissions as was seen on the previous dialog box—Allow inheritable permissions from parent to propagate to this object. When selected, this check box enables permissions changes to the parent object to affect this object. The second check box appears on folder dialog boxes only and states: "Reset permissions on all child objects and enable propagation of inheritable permissions." This control resets child object inheritance settings to their defaults.



Permission Entries:

Type	Name	Permission	Apply to
Allow	Administrators (W2KPR...	Full Control	This folder, subfolders and files
Allow	CREATOR OWNER	Full Control	Subfolders and files only
Allow	Power Users (W2KPRO...	Modify	This folder, subfolders and files
Allow	SYSTEM	Full Control	This folder, subfolders and files
Allow	Users (W2KPRO\Users)	Read & Exec...	This folder, subfolders and files

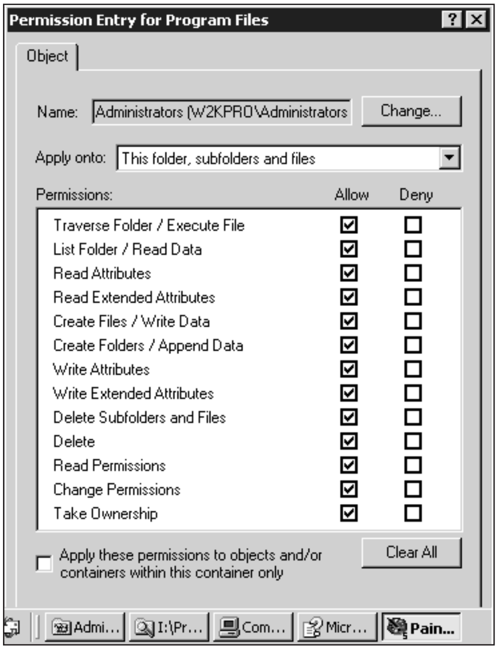
This permission is defined directly on this object. This permission is inherited by child objects.

□ Allow inheritable permissions from parent to propagate to this object

□ Reset permissions on all child objects and enable propagation of inheritable permissions.

Figure 4-16 The Access Control Settings dialog box, Permissions tab

To edit the permissions of a user or group, select them from the list and click View/Edit. The Permission Entry dialog box (see Figure 4-17) is displayed and shows all of the permissions specific to the user or group. It also has the familiar Allow and Deny check boxes.



Permission Entry for Program Files

Object

Name: Administrators (W2KPRO\Administrators

Apply onto: This folder, subfolders and files

Permissions:	Allow	Deny
Traverse Folder / Execute File	☑	□
List Folder / Read Data	☑	□
Read Attributes	☑	□
Read Extended Attributes	☑	□
Create Files / Write Data	☑	□
Create Folders / Append Data	☑	□
Write Attributes	☑	□
Write Extended Attributes	☑	□
Delete Subfolders and Files	☑	□
Delete	☑	□
Read Permissions	☑	□
Change Permissions	☑	□
Take Ownership	☑	□

□ Apply these permissions to objects and/or containers within this container only

Figure 4-17 The Permission Entry dialog box

The detailed NTFS object permissions are:

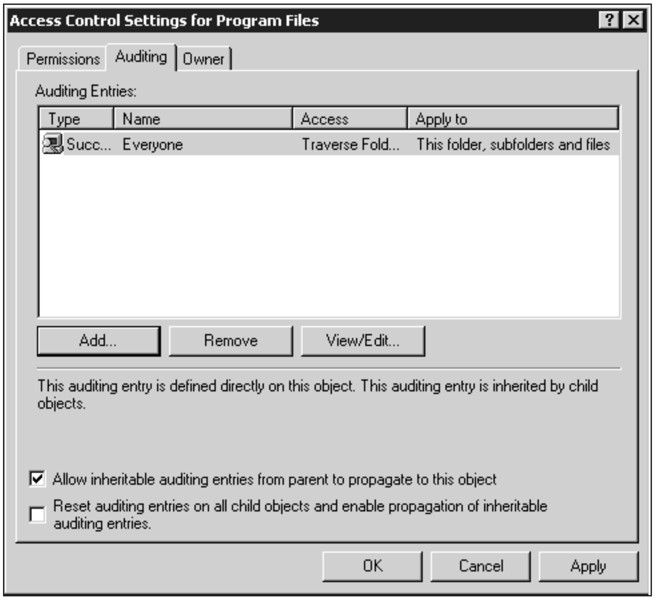
- *Traverse Folder/Execute File*—Users can see the directory structure and execute files within that structure.
- *List Folder/Read Data*—Users can view and read folders and the data they contain.
- *Read Attributes*—Users can read data.
- *Read Extended Attributes*—Users can read extended permissions. Extended attributes are defined by programs and may vary from program to program.
- *Create Files/Write Data*—The Create Files portion is a directory level permission that allows new files to be created within a directory. The Write Data permission is for files that allow a user to overwrite the existing file.
- *Create Folders/Append Data*—Users can create folders and append data to the structure.
- *Write Attributes*—Users can write data to files and folders.
- *Write Extended Attributes*—Users can change extended attributes added to objects by programs.
- *Delete Subfolders and Files*—Users can delete subfolders and files.
- *Delete*—Users can delete files and folders.
- *Read Permissions*—Users can read a file.

- *Change Permissions*—Users can change permissions of an object.
- *Take Ownership*—Administrators can acquire ownership of an object.

The Permission Entry dialog box allows you to:

- Change the user or group to which these settings apply (only on objects that do not inherit their permissions)
- (Folders only) set the application of these permissions to: this folder only; this folder, subfolder, and files; this folder and subfolders; this folder and files; subfolders and files only; subfolders only; or files only
- Clear all Allow and Deny check boxes
- Apply these permissions to objects and/or containers within this container only (folders only)

Auditing is recording the occurrence of defined system events or actions. The Auditing tab on the Access Control Settings dialog box (see Figure 4-18) is used to define events that result in an audit detail being written to the Event Viewer's Security log. This tab functions the same way as the Permissions tab. Two check boxes regarding inheritance appear at the bottom, but they apply to audit settings. Users and groups are included or deleted with the Add and Remove buttons. Selected users and groups are edited with the View/Edit button. This button reveals a similar dialog box with all of the detailed permissions. Selecting Allow or Deny on this dialog box indicates that when a user or group performs this action, an audit detail will be written to the Event Viewer Security log.



Auditing Entries:

Type	Name	Access	Apply to
Succ...	Everyone	Traverse Fold...	This folder, subfolders and files

This auditing entry is defined directly on this object. This auditing entry is inherited by child objects.

☑ Allow inheritable auditing entries from parent to propagate to this object

□ Reset auditing entries on all child objects and enable propagation of inheritable auditing entries.

Figure 4-18 The Access Control Settings dialog box, Auditing tab

The Owner tab lists the current owner of the object. To change ownership, select a new owner from the list of possible owners in the center field. The center field lists your user account and group memberships (which have Take Ownership permissions on this object). It also has a check box that can be used to replace the ownership on all child elements with the settings on this object (folders only).

NTFS Permission Rules

There are a few rules to keep in mind when working with NTFS permissions:

- NTFS object permissions *always* apply, regardless of whether the accessing user is local or remote (that is, over a network via a share).
- NTFS object permissions are cumulative. All user-specific permissions are added to all group-specific memberships (assuming the user account is a member of that group). The resulting accumulation of permissions is the access level granted to the user.
- NTFS file permissions override any contradictory settings on the parent or container folder.
- Deny overrides all other specific Allows. That is, if a user is allowed one permission, but is also assigned Deny, then the user is *denied* access regardless of the Allowed access.
- When disabling inheritance for an NTFS object, you will need to select to either Copy the parent's permissions to the current object or Remove permissions assigned from the parent and retain only object-specific settings. In either case, Copy or Remove, all subsequent changes to the parent will not affect the child object.

Copying and Moving NTFS Objects

Copying and moving NTFS objects is an important subject because of the inheritance of permissions. When a new object is created, it always assumes the permissions (and other settings and attributes) of its parent or container. Keep this in mind to help you understand what happens when an NTFS object is copied or moved. There are four different moving/copying scenarios when dealing with NTFS source and destination volumes or partitions:

- Moving an object within the same volume or partition
- Copying an object within the same volume or partition
- Moving an object from one volume or partition to another
- Copying an object from one volume or partition to another

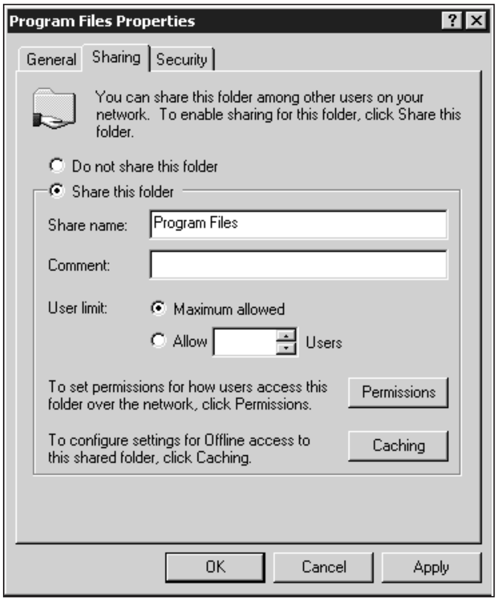
Moving an object within the same volume or partition is actually just a minor change in the location pointer for the object. Thus, its new location is not caused by creating a new file, but just by changing its location address. Such objects retain their original NTFS permissions.

All of the other copy and move situations involve creating a new object. This is obvious for the copy procedure, but when moving from one volume to another, a two-step process is used. First, the system copies the file to the new destination. Second, it deletes the original. The act of creating a new object causes that new object to inherit the NTFS permissions of its new parent or container.

When moving or copying an object from an NTFS volume to a FAT volume, all NTFS settings are lost and the object inherits the FAT attributes and settings of its new container. When moving or copying an object from a FAT volume to an NTFS volume, the object inherits the NTFS settings and permissions of its new container.

MANAGING SHARED FOLDERS

The Sharing tab (see Figure 4-19) found on both FAT/FAT32 and NTFS folder Properties dialog boxes is used to enable remote access to the folder. This tab is used to share and configure sharing for this object. Selecting Share this folder or Do not share this folder either offers the resource to the network or removes the share.



You can share this folder among other users on your network. To enable sharing for this folder, click Share this folder.

○ Do not share this folder

● Share this folder

Share name: Program Files

Comment:

User limit: ● Maximum allowed

○ Allow Users

To set permissions for how users access this folder over the network, click Permissions.

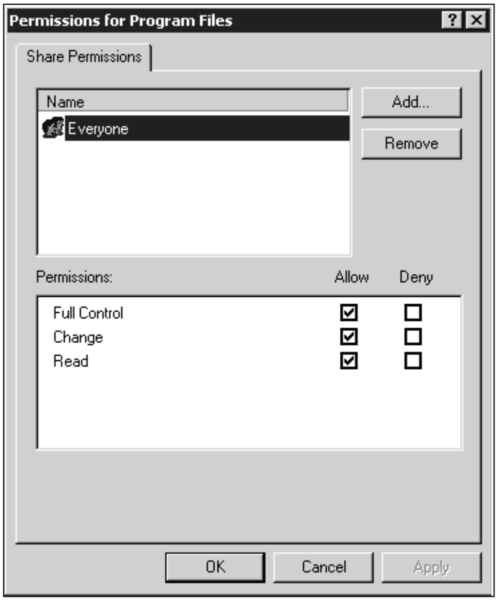
To configure settings for Offline access to this shared folder, click Caching.

Figure 4-19 A folder's Properties dialog box, Sharing tab

The Sharing tab offers the following controls:

- *Do not share this folder*—Disables sharing for this folder
- *Share this folder*—Enables sharing for this folder
- *Share name*—The name displayed in browse lists and used in UNC (Universal Naming Convention) names to access this share

- *Comment*—A comment about or description of the share
- *User limit*—Used to allow the maximum possible users (as determined by system speed and resources), or to limit simultaneous users to a specified number
- *Permissions button*—Opens the Share Permissions dialog box (see Figure 4-20) where users and groups are granted or denied Full Control, Change, or Read permissions for this folder via the share
- *Caching button*—Opens the Caching Settings dialog box where you can enable or disable caching of resources from this folder and set caching to automatic for documents or programs, or manual for documents. This feature is used in conjunction with the Offline Files settings of Folder Options to cache network resources for use while not connected to the network.



Name: Everyone

Permissions:	Allow	Deny
Full Control	☑	□
Change	☑	□
Read	☑	□

Figure 4-20 The Share Permissions dialog box

The three share permission levels are:

- *Read*—Allows users to access, execute, and open resources via the share
- *Change*—Allows users to create new objects, change and delete existing objects, and to access, execute, and open resources via the share
- *Full Control*—Allows users to perform all actions on resources via the share

There are several important issues to keep in mind when working with shares:

- The three permission levels on a share are the only way to impose security on shared FAT volumes.
- Shares are folders, not individual files.

- Share permissions apply only to the network access point.
- The default permission of a share is Full Control allowed for the Everyone group.
- Multiple share permission levels because of group memberships are cumulative.
- Deny always overrides any other specific Allow.
- The most restrictive permissions of cumulative share and cumulative NTFS apply.
- Share permissions only restrict access for network users, not local users.
- A moved folder is no longer shared.
- A copied folder is not shared, but the original folder retains its shared status.

TIP

Shared folders are easy to recognize because their folder icon has a blue-sleeved hand supporting the folder.

CAUTION

Because the default permission of a share is Full Control allowed for the Everyone group, be sure to set NTFS permissions for shares as they are created to avoid a security breach.

Accessing shared resources on a Microsoft network is handled through several mechanisms. You can map a drive using the Map Network Drive command from Windows Explorer (try Hands-on Project 4-15), My Computer, or My Network Places. You can access shared resources via the My Network Places tool. Most Open and Save dialog boxes offer a link to My Network Places allowing you to open or save files to remote paths. You can also access shares via UNC paths using the Run command. If you use My Network Places, you can use the following access methods:

- *Add Network Place*—A wizard used to map a share to My Network Places (does not assign a drive letter to the mapped share).
- *Computers Near Me*—Lists all computers in your domain or workgroup. Each of these can be accessed to reveal shared resources.
- *Entire Network*—Lists all domains or workgroups seen on the network. Each of these can be accessed to see members of those domains or workgroups. Each of these members can be accessed to reveal shared resources.

TIP

Windows 2000 domains are listed under the Directory heading (for Active Directory). Windows NT/95/98 domains and workgroups and Windows 2000 workgroups are listed under the Microsoft Windows Network heading.

TROUBLESHOOTING PERMISSIONS PROBLEMS

In most cases of access problems, one of two situations exists: either the resource object has the wrong settings, or the user account has the wrong settings. A resource object can have incorrect permissions settings due to inheritance, lack of inheritance, moving/copying, or simple human error (for example, setting the wrong thing). A user account can have the incorrect permissions due to improper group membership, improper permission settings on a valid group, or human error.

To resolve permission or access problems, follow this procedure:

1. Determine what valid access the user should have.
2. Inspect the resource object's permissions based on groups and the specific user and what actions are set to Allow or Deny.
3. Inspect the share's permissions based on groups and the specific user and what actions are set to Allow or Deny.
4. Inspect the user's group memberships.
5. Attempt to access other resources with the user account from the same computer and from a different computer.
6. Attempt to access the problematic resource with the Administrator account from the same computer and from a different computer.

The preceding steps should point you directly to the problem and how to resolve it. Taking the time to make the effort systematic will prevent you from overlooking small details or glaringly obvious problems.

In general, use the following guidelines to lay out or design permission levels to avoid common problems:

- Grant permission only as needed.
- Rely upon NTFS to restrict access.
- Grant Full Control only when necessary, even on shares.
- Change permissions on a folder level, and allow changes to affect all child elements (at least to files, if not subfolders).
- Use multiple folders and subfolders to separate files into groups for different permission levels.
- Do not use the Deny setting unless absolutely necessary.

Optimizing access to files and folders requires a two-part verification. The first verification is to ensure that the share- and direct object-level permissions grant and restrict exactly the activities you want for each user and group. The second verification is to ensure that group memberships do not grant too much access via accumulated rights or prevent necessary access due to a specific right or permission having the Deny box checked. Both of these verification processes must be performed manually.

THE MICROSOFT DISTRIBUTED FILE SYSTEM

The Microsoft **Distributed File System (DFS)** is a Windows 2000 Server hosted service used to manipulate and manage shared resources. DFS combines shared resources from various locations throughout a network into a single hierarchical system. This allows DFS to be a single access or reference point for a logical tree structure without regard to the physical location of the resources. DFS functions by first creating a DFS root on a Windows 2000 Server system. This root looks and acts much like a share. Then shared resources from other systems can be mapped under the DFS root. These are called DFS child nodes. The DFS child nodes appear as subfolders underneath the DFS root.

The benefits of DFS include:

- All network resources are organized in a single tree structure.
- User navigation of resources is simplified because the host computer name is not required.
- Simplified administration. If a server that hosts resources fails, the path to a new alternate location can be defined without affecting the path employed by users to gain access.
- Access permissions are preserved.
- The DFS root is accessed in the same way as a normal share.
- Once inside the DFS root, all other resource accesses are simplified and do not require knowing the name of the host systems.

From a client's perspective, DFS provides simplified access to all resources in an enterprise. Special client software is required to use DFS. Windows 2000 Professional, Windows 95 and 98, Windows for Workgroups, and Windows NT all include this software.

For more information on DFS, see Windows 2000 Server documentation and the *Windows 2000 Server Resource Kit*.

CHAPTER SUMMARY

- ❐ This chapter discussed dynamic storage, the new Windows 2000 storage mechanism that does not rely upon partitions as does the basic storage method. The divisions of dynamic storage are called volumes. All volumes and partitions can be formatted with FAT, FAT32, or NTFS. Preexisting basic storage drive configurations can be managed by Windows 2000, but only dynamic devices can be used to create new multipart drive configurations. Basic storage devices can be converted to dynamic devices without damaging the data, but to reverse the process requires that all volumes be deleted before converting back to basic storage. The Disk Management snap-in is used to perform all drive-, partition-, and volume-related functions. Windows 2000 supports simple volumes, spanned volumes, striped volumes, mirrored volumes, and RAID 5 volumes.

- The FAT and FAT32 file systems are retained by Windows 2000 for backward compatibility with other operating systems on the same multiboot system. FAT does not offer any form of file level security. NTFS is the recommended file system to use under Windows 2000 because it offers file level security, encryption, and disk quotas.

- Mount points are a new mapping method in Windows 2000. This method allows volumes or partitions to be mapped to empty directories on NTFS volumes or partitions. With map points, up to 32 drives can be utilized within the limitation of drive letters. There are several disk-related utilities: Disk Cleanup, ScanDisk, and Disk Defragmenter.

- All file system objects within Windows 2000 have unique properties and controls. Plus, all NTFS objects offer security, encryption, compression, and auditing. NTFS permissions are used to control access to resources. Shares are used to grant access to local resources across a network.

- The Microsoft Distributed File System (DFS) is a service used to manipulate and manage shared resources. DFS can combine shared resources from throughout an enterprise into a single hierarchical system.

KEY TERMS

active partition — A primary partition is marked active when it hosts the necessary files to boot into an operating system.

auditing — The recording of the occurrence of a defined event or action.

basic storage — The drive division method that employs partitions.

boot partition — The partition that hosts the main Windows 2000 system files and is the initial default location for the paging file. The boot partition can be the same partition as the system partition, or it can be any other partition (or logical drive in an extended partition) on any drive hosted by the computer.

cluster — One or more sectors grouped into a single nondivisible unit.

defragmentation — The process of reorganizing files so they are stored contiguously and no gaps are left between files.

Disk Management — The Microsoft Management Console (MMC) snap-in used to manage drives.

disk quota — A limitation on the amount of disk space that can be consumed by a user.

Distributed File System (DFS) — A Windows 2000 Server hosted service used to manipulate and manage shared resources from various locations throughout a network in a single hierarchical system.

drive letter — One of two methods of accessing file system resources on formatted volumes under Windows 2000. A drive letter can be assigned to a partition or volume or a drive configuration of multiple components.

dynamic storage — The drive division method that employs volumes. It is a new standard supported only by Windows 2000.

extended partition — A type of partition on a basic disk that can be divided into logical drives. Only a single extended partition can exist on a physical disk, and when present only three primary partitions can exist.

FAT (FAT16) — The 16-bit file allocation table file system originally introduced with DOS. As supported under Windows 2000, it can be used to format partitions or volumes up to 4 GB.

FAT32 — The 32-bit FAT file system. As supported under Windows 2000, it can be used to format partitions or volumes up to 32 GB.

fragmentation — The division of a file into two or more parts where each part is stored in a different location on the hard drive. As the level of fragmentation on a drive increases, the longer it takes for read and write operations to occur.

long filenames (LFNs) — Filenames up to 256 characters in length, supported by all file systems under Windows 2000.

mirrored volume — A drive configuration of a single volume is duplicated onto another volume on a different hard drive and provides fault tolerance. In Windows NT, a mirror onto a drive hosted by a different drive controller was called duplexing, but this distinction is no longer used in Windows 2000 (Windows 2000 Server only).

mount point or **mounted volume** — A new drive access technique that maps a volume or partition to an empty directory on an NTFS volume or partition.

NTFS (New Technology File System) — The preferred file system of Windows 2000. Supports file level security, encryption, compression, auditing, and more. Supports volumes up to 2 TB.

partition — A logical division of the physical space on a hard drive.

POSIX (Portable Operating System Interface for Computing Environments) — A set of standards drafted by the Institute of Electrical and Electronic Engineers (IEEE) that defines various aspects of an operating system, including topics such as programming interface, security, networking, and graphical interface.

primary partition — A type of partition on a basic disk that can be marked active. Up to four primary partitions can exist on a physical disk, but only one partition can be active.

RAID 5 volume — A drive configuration of three or more (up to 32) parts of one or more drives or three or more (up to 32) entire drives. Data is written to all drives in equal amounts to spread the workload. Parity information is added to the written data to allow for drive failure recovery. Provides fault tolerance. If one partition or drive fails in the set, the other members can re-create the missing data on the fly. When the failed member is replaced or repaired, the data on that drive can be rebuilt and restored. This is also known as disk striping with parity (Windows 2000 Server only).

removable storage device — Any type of floppy, cartridge, or drive that can be either removed between reboots or as a hot swappable device.

sector — The smallest division (512 bytes) of a drive's surface.

share — A resource, such as an application, file, printer, or other device, that can be accessed over the network.

simple volume — A drive configuration of all or part of a single drive. Does not provide any fault tolerance. NTFS volumes can be extended; FAT and FAT32 volumes cannot be extended.

spanned volume — A drive configuration of two or more (up to 32) parts of one or more drives or two or more entire drives; the elements of the spanned volume do not have to be equal in size.

striped volume — A drive configuration of two or more (up to 32) parts of one or more drives or two or more (up to 32) entire drives. Data is written to all drives in equal amounts (in 64 KB units) to spread the workload and improve performance.

system partition — The active partition where the boot files required to display the boot menu and initiate the booting of Windows 2000 are stored.

volume — (1) In basic storage, a collection of 2 to 32 partitions into a single logical structure. (2) In dynamic storage, any division of a physical drive or collection of divisions into a drive configuration.

volume set — A collection of disk partitions that are treated as a logical drive. A volume set may be expanded after it has already been created. To make a volume set smaller, however, you must back up all the data, delete the volume set, define a new (smaller) volume set, and restore the data to that set. If you lose one drive in a volume set, you lose all the data in the entire set, because it offers no fault tolerance.

REVIEW QUESTIONS

1. Which storage method employs primary and extended partitions?
   a. logical drives
   b. basic
   c. dynamic
   d. spanned volumes
2. When logical drives are present on a basic storage device, how many primary partitions can exist?
   a. 1
   b. 2
   c. 3
   d. 4
3. Which of the following statements are true about a volume set, comprised of either partitions or volumes? (Choose all that apply.)
   a. combines two or more volumes/partitions into a single logical storage area
   b. provides fault tolerance
   c. If one element of the set fails, all data in the set is lost.
   d. It can be assigned a single drive letter.

4. A 4 GB partition or volume can be formatted with what file system? (Choose all that apply.)
   a. FAT
   b. FAT32
   c. HPFS
   d. NTFS
5. What mechanism(s) of Windows 2000 allow you to access up to 32 volumes on a single system?
   a. shares
   b. drive letters
   c. DFS
   d. mounted volumes
6. Under Windows 2000, it is possible to create new RAID 5 volumes on dynamic and basic drives. True or False?
7. Which of the following is the partition that hosts the main Windows 2000 system files and is the initial default location for the paging file?
   a. system partition
   b. boot partition
   c. logical partition
   d. dynamic partition
8. The drive configurations supported by Windows 2000 Professional provide fault tolerance. True or False?
9. What is the best file system for a 250 MB volume?
   a. FAT
   b. FAT32
   c. NTFS
10. NTFS volumes created under Windows 2000 cannot be accessed by any other operating system. True or False?
11. Which of the following are true for NTFS under Windows 2000? (Choose all that apply.)
    a. supports volumes up to 2 TB in size
    b. file level compression, encryption, auditing, and security
    c. disk quotas
    d. POSIX file system support
    e. most efficient on volumes smaller than 512 MB
12. Drives can be converted to and from dynamic storage without damaging the hosted data. True or False?

13. The Properties dialog box for a partition or volume gives you quick access to which drive tools? (Choose all that apply.)
    a. ScanDisk
    b. Defragmentation
    c. Disk Cleanup
    d. Device Manager
    e. Backup
    f. EventViewer
14. A volume or partition can be attached to a mount point on any other volume or partition. True or False?
15. Quotas can be defined in what manner(s)?
    a. by user
    b. by drive
    c. by group
    d. by volume or partition
16. Disk Cleanup is used to free space on a hard drive by removing orphaned files, cleaning out the Recycle bin, and shrinking the page file. True or False?
17. The built-in defragmentation utility can be scheduled to automatically reorganize local hard drives. True or False?
18. Which of the following are properties of NTFS file or folder objects, but not of FAT file or folder objects? (Choose all that apply.)
    a. Attributes: Read-only
    b. compress contents to save disk space
    c. Attributes: Archive
    d. encrypt contents to secure data
19. What methods can be used to prevent a user from gaining access to an NTFS resource?
    a. Do not include the user account (or its groups) in the list of permissions.
    b. Set the user account's permissions to Deny.
    c. Set the user account's permissions to No Access.
    d. Place the user account in the Guests group.
20. NTFS object permissions are used only when a user is local. True or False?
21. Which of the following are true?
    a. Child objects can inherit the permissions of their parent containers.
    b. Copied files always retain their original settings.
    c. File-level permissions always override contradictory settings on the parent container.
    d. Deny allows overrides of all other specific Allows.

22. Files moved from an NTFS volume to a FAT volume and then to another NTFS volume will reassume their original settings. True or False?
23. Which of the following statements is true about shares? (Choose all that apply.)
    a. offers only three levels of permissions
    b. can be cached on client systems
    c. can restrict simultaneous users
    d. can be individual files or folders
    e. overrides NTFS permissions
    f. The most restrictive permissions of cumulative shares and cumulative NTFS apply.
24. To grant varying levels of access within a share, use NTFS permissions and group files into subfolders. True or False?
25. Which of the following statements is true about the Microsoft Distributed File System? (Choose all that apply.)
    a. All network resources are organized in a single tree structure.
    b. Access permissions are preserved.
    c. A DFS root can be hosted by Windows 2000 Professional.
    d. Once inside the DFS root, all other resource accesses are simplified and do not require knowing the name of the host systems.

HANDS-ON PROJECTS

Project 4-1

To create a partition on a basic drive:

> TIP This hands-on project requires that a basic drive with unallocated space be present in the system. Additionally, the drive must have either only three primary partitions or only two primary partitions if an extended partition is present.

1. Open the **Control Panel** (**Start, Settings, Control Panel**).
2. Open the **Administrative Tools** applet (double-click its icon).
3. Open the **Computer Management** tool (double-click its icon).
4. Expand the **Storage** console node if necessary (click on the plus sign to the left of the node).
5. Select **Disk Management**.
6. Right-click over an unallocated area of a basic drive and select **Create Partition** from the resulting menu.
7. The **Create Partition Wizard** launches. Click **Next**.

8. Select **Primary Partition**. Click **Next**.
9. Select the amount of space to use in this partition. Accept the default of the maximum space available. Click **Next**.
10. Assign a drive letter. Accept the default. Click **Next**.
11. Select the file system to format this partition. Accept the default of **NTFS**. Click **Next**.
12. The wizard displays a list of the actions to be performed in creating this partition. Click **Finish**.
13. The system will create the partition, format the drive, and assign the drive letter. The display of the drive will be updated to reflect the new partition.

Project 4-2

To change a drive letter on a volume or partition:

1. Right-click a partition or volume. Be sure not to select the boot or system partition; select **Change Drive Letter and Path** from the resulting menu.
2. Check that the current drive letter is selected, and then click **Edit**.
3. Select the **Assign a drive letter** radio button if it is not already selected.
4. Use the pull-down list to select a different letter for this drive.
5. Click **OK**.
6. You'll be warned about changing drive letters. Click **Yes**. The Disk Management display will reflect the drive letter change.

Project 4-3

To create a mounted volume:

> TIP This hands-on project requires that at least two partitions be present on the system. Partition A must be formatted with NTFS. Partition B can be any partition other than the boot or system partitions.

1. Locate Partition A. Take note of its drive letter: ________________
2. Right-click Partition B, select **Change Drive Letter and Path** from the resulting menu.
3. Click **Add**.
4. Make sure the **Mount in this NTFS folder** option is selected.
5. Click **Browse**.
6. Locate Partition A by using its drive letter (see step 1). Select the drive letter.
7. Click **New Folder**.
8. Type in a name for the new folder, such as **MapPartB**, and then press **Enter**.
9. Make sure the newly created folder is highlighted, and then click **OK**.

10. The path to the new folder is now listed in the text field under the **Mount in this NTFS folder** option. Click **OK**.
11. Open **Windows Explorer** (**Start**, **Programs**, **Accessories**, **Windows Explorer**).
12. Expand My Computer.
13. Expand Partition A.
14. Notice that the mounted volume appears as a drive icon with the name of the folder you created. Select the mount point. Notice that the contents of Partition B are displayed in the right pane.
15. Close Windows Explorer.

Project 4-4

To delete a mounted volume:

TIP This hands-on project requires that Hands-on Project 4-3 be completed.

1. Right-click over Partition B from Hands-on Project 4-3, select **Change Drive Letter and Path** from the resulting menu.
2. Select the mounted volume mapping.
3. Click **Remove**.
4. You'll be asked to confirm the deletion. Click **Yes**.

Project 4-5

To delete a partition from a basic drive:

TIP This hands-on project requires that Hands-on Project 4-1 be completed.

1. Select the partition you created in Hands-on Project 4-1.
2. Right-click and select **Delete Partition** from the resulting menu.
3. To confirm the deletion, click **Yes**.

Project 4-6

To convert a basic drive to a dynamic drive:

TIP This hands-on project requires that a second hard drive that is currently a basic drive be present on the system.

1. Select a basic disk in **Disk Management**. The selected disk cannot host the system partition.
2. Right-click and select **Upgrade to Dynamic Disk** from the resulting menu.
3. A list of all hard drives present on the system is displayed. The disk you selected will already be checked. Do not change the status of the check boxes on this list. Click **OK**.
4. Disk Management will display the drive as Dynamic.

Project 4-7

To create a volume on a dynamic drive:

TIP This hands-on project requires that Hands-on Project 4-6 be completed.

1. Right-click over the unallocated space on a dynamic drive, select **Create Volume** from the resulting menu.
2. This launches the **Create Volume Wizard**. Click **Next**.
3. Select the volume type to create. In this case, select **Simple volume**. Click **Next**.
4. For a simple volume you only need unallocated space from a single drive. Make sure the drive is listed in the Selected dynamic disk field.
5. In the **Size** field, enter the amount of space from the maximum available to use in the volume. Enter an amount of about one-half of that available. Click **Next**.
6. Assign a drive letter. Accept the defaults. Click **Next**.
7. Select the file system to format the new volume. Accept the default of NTFS. Click **Next**.
8. The wizard displays a list of actions to be performed in creating the new volume. Click **Finish**.

Project 4-8

To extend a volume:

This hands-on project requires that Hands-on Project 4-7 be completed.

1. Right-click over the volume created in Hands-on Project 4-7, and then select **Extend Volume** from the resulting menu.
2. The Extend Volume Wizard is launched. Click **Next**.
3. Make sure the drive with unallocated space is listed in the Selected dynamic disk field.
4. Change the size of the remaining unallocated space to be added to the existing volume to 80% of the space available (for example, if there are 200 MB remaining, change the number to 180). Click **Next**.
5. The wizard displays a list of actions to perform in extending the volume. Click **Finish**.
6. Disk Management will display the extension with the same drive letter as the original volume.

Project 4-9

To delete a volume:

This hands-on project requires that Hands-on Project 4-7 be completed.

1. Select the volume you created in Hands-on Project 4-7.
2. Right-click and select **Delete Volume** from the resulting menu.
3. Confirm the deletion. Click **Yes**.
4. Disk Management displays the drive as hosting no volumes and consisting only of unallocated space.

Project 4-10

To revert a dynamic drive to a basic drive:

> TIP This hands-on project requires that Hands-on Project 4-9 be completed.

1. Select the drive used in Hands-on Projects 4-6 through 4-9.
2. Right-click and select **Revert to Basic Disk** from the resulting menu.
3. Disk Management will display the drive as Basic.

Project 4-11

To compress and decompress a folder:

1. Launch Windows Explorer (**Start, Programs, Accessories, Windows Explorer**).
2. Locate and select any folder on your hard drive, such as C:\Program Files.
3. Right-click the folder, then select **Properties** from the menu.
4. On the **General** tab, take note of the Size and Size on Disk values.
5. On the **General** tab, click the **Advanced** button.
6. Select the **Compress contents to save disk space** checkbox.
7. Click **OK**.
8. Click **OK**.
9. Select the **Apply Changes to this folder**, **subfolder**, **and files** radio button.
10. Click **OK**. The system will compress the folder and its contents, this may take several minutes.
11. Right-click the folder, then select **Properties** from the menu.
12. On the **General** tab, take note of the Size and Size on Disk values. The Size on Disk value should be smaller than the original value.
13. On the **General** tab, click the Advanced button.
14. Deselect the **Compress contents to save disk space** checkbox.
15. Click **OK**.
16. Click **OK**.
17. Select the **Apply Changes to this folder**, **subfolder**, **and files** radio button. Click **OK**. The system will decompress the folder and its contents, this may take several minutes.

Project 4-12

To optimize folder access:

This hands-on project requires that Windows 2000 be installed and an NTFS partition is present.

1. Launch Windows Explorer (**Start, Programs**, **Accessories**, **Windows Explorer**).
2. In the left pane, select a drive formatted with NTFS within My Computer.
3. In the right pane, select a file or folder.
4. From the **File** menu, select **Properties**.
5. Select the **Security** tab.
6. Click the **Add** button.
7. Select the **Authenticated Users** group.
8. Click **Add**.
9. Click **OK**.
10. Click the **Authenticated Users** group, which now appears in the list of names on the Security tab for the NTFS object. Take note of the granted permissions.
11. Select the **Sharing** tab.
12. Select the **Share this Folder** radio button.
13. Click **Permissions**.
14. Click **Add**.
15. Select the **Authenticated Users** group.
16. Click **Add**.
17. Click **OK**.
18. Set the Share permissions for the Authenticated Users group as close to the NTFS file level permissions as possible.
19. Click **OK**.
20. Click **OK**.

Project 4-13

To share a folder and remove a share:

TIP This hands-on project requires that Windows 2000 be installed and an NTFS partition is present.

1. Launch Windows Explorer (**Start, Programs**, **Accessories**, **Windows Explorer**).
2. In the left pane, select a drive formatted with NTFS within My Computer.
3. In the right pane, select a file or folder.
4. From the **File** menu, select **Sharing**.
5. Select the **Share this folder** radio button.
6. Click the **Permissions** button.
7. Click **Add**.
8. Select the **Authenticated Users** group.
9. Click **Add**.
10. Click **OK**.
11. Click **OK**. Notice the folder now has a shared hand on its icon.
12. With the folder still selected, select the **Sharing** command from the **File** menu.
13. Select the **Do not share this folder** radio button.
14. Click **OK**. Notice the shared hand on the folder icon disappears.

Project 4-14

To convert a FAT partition to NTFS:

CAUTION

This hands-on project requires that a FAT volume exist on your Windows 2000 system. This volume will be converted to NTFS. Proceed only if the conversion of this volume will not compromise your system.

1. Launch the Command Prompt by selecting **Start**, **Programs**, **Accessories**, **Command Prompt**.
2. Change to the FAT partition drive, such as by typing **g:**, then press **Enter**.
3. Type **convert g: /fs:ntfs /v** where g: is the drive letter of the FAT volume to convert, press **Enter**.

4. Provide the current label for the drive to be converted, look in **Windows Explorer** to see what the volume name is. Press **Enter** after you have typed in the volume label.
5. You'll be prompted whether to complete the conversion at the next reboot, press **Y**.
6. Reboot the computer. The drive will be converted as part of the startup process.

Project 4-15

To map to a network drive:

TIP This project requires that the Windows 2000 Professional be a client on a network with at least one shared folder available for mapping.

1. Launch Windows Explorer (**Start, Programs**, **Accessories**, **Windows Explorer**).
2. Select the **Map Network Drive** command from the **Tools** menu.
3. Click the **Browse** button.
4. Using the browse list, locate and select a shared folder from the network.
5. Click **OK**.
6. Select a drive letter using the pull-down list next to **Drive:**.
7. Deselect the **Reconnect at logon** checkbox.
8. Click **Finish**.

Case Projects

1. You must test new media software that plays large multimedia presentations (often 3 MB or larger). The software is being developed for Windows 98, Windows NT, and Windows 2000. Can you configure a multiboot system with all three operating systems in such a way that a single drive can host at least six media presentations which can be accessed from all three OSs? If so, how? If not, what other solution(s) can be used?
2. The security requirements of your organization state that log files of system access must be retained for at least six months on live accessible drives. In the past, these log files have consumed at least 6 GB of drive space per month. However, they are growing larger at an accelerated rate. Because you don't know exactly how much drive space you'll need over the next year or even six months, what options do you have under Windows 2000 to comply with the organization's security requirements?